

COLÉGIO EVANGÉLICO PANAMBI

CURSO TÉCNICO EM INFORMÁTICA

RAFAEL MATEUS DALLABRIDA

**SISTEMA DE CONTROLE DE FINANÇAS**

**E OBJETIVOS FINANCEIROS**

PANAMBI

2024

RAFAEL MATEUS DALLABRIDA

**SISTEMA DE CONTROLE DE FINANÇAS**

**E OBJETIVOS FINANCEIROS**

Projeto de conclusão de curso apresentado como requisito parcial para obtenção do título de Técnico em Informática, do Curso de Técnico em Informática do Colégio Evangélico Panambi – Centro Tecnológico e de Formação Profissional.

Orientadores: Prof. Catana

Prof. Rodrigo Steinhorst

PANAMBI

2024

RAFAEL MATEUS DALLABRIDA

**SISTEMA DE CONTROLE DE FINANÇAS**

**E OBJETIVOS FINANCEIROS**

Projeto de conclusão de curso apresentado como requisito parcial para obtenção do título de Técnico em Informática, do Curso de Técnico em Informática do Colégio Evangélico Panambi – Centro Tecnológico e de Formação Profissional.

Banca Examinadora

Prof. Rodrigo Steinhorst

Prof. Catana Rahmeier

Prof. Sirlei Adriane Wottrich

PANAMBI

2024

Sumário

[INTRODUÇÃO 6](#_Toc170385892)

[GOALSPLAN 7](#_Toc170385893)

[CARACTERÍSTICAS DO SISTEMA 8](#_Toc170385894)

[1. Funcionalidades: 8](#_Toc170385895)

[2. Tecnologias utilizadas: 8](#_Toc170385896)

[3. Resultados Esperados 9](#_Toc170385897)

[MODELO CONCEITUAL 10](#_Toc170385898)

[DICIONÁRIO DE DADOS 12](#_Toc170385899)

[UTILIZANDO O SISTEMA 15](#_Toc170385900)

[1. Login 15](#_Toc170385901)

[2. Padrão de Cadastro 16](#_Toc170385902)

[3. Página Inicial 17](#_Toc170385903)

[4. Lista de Registros 18](#_Toc170385904)

[CONCLUSÃO 19](#_Toc170385905)

[REFERÊNCIAS 20](#_Toc170385906)

[FERRAMENTAS UTILIZADAS 22](#_Toc170385907)

[CÓDIGO DO BANCO DE DADOS 23](#_Toc170385908)

**Índice de Imagens**

[Figura 1 - Modelo Conceitual Parte 1 10](https://d.docs.live.net/2241cfda359a2025/Documentos/PROJETO-CONCLUSAO-CURSO.docx#_Toc170353983)

[Figura 2 - Modelo Conceitual Parte 2 11](https://d.docs.live.net/2241cfda359a2025/Documentos/PROJETO-CONCLUSAO-CURSO.docx#_Toc170353984)

[Figura 3 - Tela de login 15](https://d.docs.live.net/2241cfda359a2025/Documentos/PROJETO-CONCLUSAO-CURSO.docx#_Toc170353985)

[Figura 4 - Erro no login 15](https://d.docs.live.net/2241cfda359a2025/Documentos/PROJETO-CONCLUSAO-CURSO.docx#_Toc170353986)

[Figura 5 - Cadastro de usuário 16](https://d.docs.live.net/2241cfda359a2025/Documentos/PROJETO-CONCLUSAO-CURSO.docx#_Toc170353987)

[Figura 6 - Página inicial 17](https://d.docs.live.net/2241cfda359a2025/Documentos/PROJETO-CONCLUSAO-CURSO.docx#_Toc170353988)

[Figura 7 - Lista de Registros 18](https://d.docs.live.net/2241cfda359a2025/Documentos/PROJETO-CONCLUSAO-CURSO.docx#_Toc170353989)

[Figura 8 - Lista de Pagamentos com checkbox 18](https://d.docs.live.net/2241cfda359a2025/Documentos/PROJETO-CONCLUSAO-CURSO.docx#_Toc170353990)

# INTRODUÇÃO

Este sistema tem como objetivo ajudar o usuário a organizar suas finanças, seus pagamentos e recebimentos, de forma que consiga categorizar seus lançamentos e ter um retorno visual de quanto recurso financeiro lhe falta para atingir seus objetivos.

O sistema é uma aplicação web que conta com páginas de cadastro e listagem de registros intuitivas e de fácil utilização e vem para suprir uma falta de sistemas similares que sejam simplificados para o uso cotidiano. Quando comparado a outros sistemas, este apresenta apenas a inclusão de dados essências para organizar finanças pessoais, que não sejam vinculadas à uma empresa, tornando essa organização menos burocrática.

Realizando os cadastros corretamente, o usuário terá fácil acesso ao valor monetário que possui e poderá acompanhar seu progresso em relação aos objetivos que deseja alcançar.

# GOALSPLAN

O sistema foi nomeado como “GoalsPlan” unindo duas palavras da língua inglesa: *Goals*, que significa “Objetivos” e *Plan*, “Planejar”. Como o próprio nome sugere, este sistema terá a função de auxiliar seus usuários a fazer o planejamento para alcançar seus objetivos (também chamados de metas), função de muita importância, principalmente nos dias de hoje, pois sempre somos bombardeados por diversos anúncios que nos instigam a cada vez mais desviar o foco de nossos objetivos e adquirir algo que muitas vezes nem se faz necessário.

Ter um controle sobre seus gastos e recebimentos é muito vantajoso para aqueles que o conseguem fazer. Porém, nem sempre é uma tarefa fácil manter um caderno ou uma planilha com essas informações. Podem ser facilmente perdidas e só podem ser acessadas com o caderno em mãos ou tendo o arquivo em seu computador. Então é uma boa ideia recorrer à um sistema online para atender a essa demanda, até mesmo por ter uma resposta mais dinâmica dessas informações.

Os sistemas que existem hoje para facilitar esse processo muitas vezes ou são pagos (e nada baratos) ou são muito complexos de se entender, fazendo a pessoa se desanimar com essa atividade que normalmente já não é a de maior interesse em sua rotina.

A *GoalsPlan* surge como uma solução à estes problemas mencionados, um sistema web de fácil acesso e de complexidade reduzida para tornar o processo de organização de finanças mais dinâmico e menos exaustivo.

# CARACTERÍSTICAS DO SISTEMA

## Funcionalidades:

* 1. Cadastro de Usuários: Permite a privacidade de dados, todas as tabelas do banco de dados possuem o usuário como chave, garantindo assim que cada usuário possa ver apenas seus próprios dados;
  2. Página Inicial: Apresenta o saldo total do usuário, bem como uma visão geral de todas as suas metas cadastradas que ainda não foram concluídas, com uma barra de progresso indicando quanto falta para atingi-lás;
  3. Cadastro de Entidades: Garante o cadastro personalizado e pessoal das seguintes entidades por usuário:
     1. Fornecedores - instituições ou pessoas a quem é feito pagamentos;
     2. Clientes - instituições ou pessoas das quais o usuário recebe;
     3. Históricos – finalidades recorrentes dos lançamentos (exemplos: sálario, lazer, educação, alimentação, investimento, etc);
     4. Metas – objetivos que se deseja alcançar (exemplos: carro, computador, casa, etc);
     5. Pagamentos: valores que saem do saldo do usuário com destino à algum fornecedor;
     6. Recebimentos: valores que acrescentam o saldo do usuário vindos de algum cliente.

## Tecnologias utilizadas:

* 1. HTML - usado para estruturar as páginas web;
  2. CSS – usado para deixar as páginas visualmente mais agradáveis e intuitivas;
  3. PHP – linguagem de programação usada para manipular os dados dos campos HTML e fazer a conexão com o banco de dados;
  4. JavaScript – linguagem de programação usada para fazer interações dinâmicas entre o sistema e o usuário;
  5. MariaDB – banco de dados de código aberto usado para armazenar e consultar os dados enviados pelas páginas web;
  6. SQL – linguagem de consulta usado para consultar, inserir, alterar e excluir dados do banco de dados.

1. Resultados Esperados:
   1. Auxiliar o usuário a registrar e organizar suas finanças;
   2. Manter o foco do usuário em relação às suas metas cadastradas apresentando sua progressão de forma visual;
   3. Tornar o processo de organização de finanças menos desgastante.

# MODELO CONCEITUAL

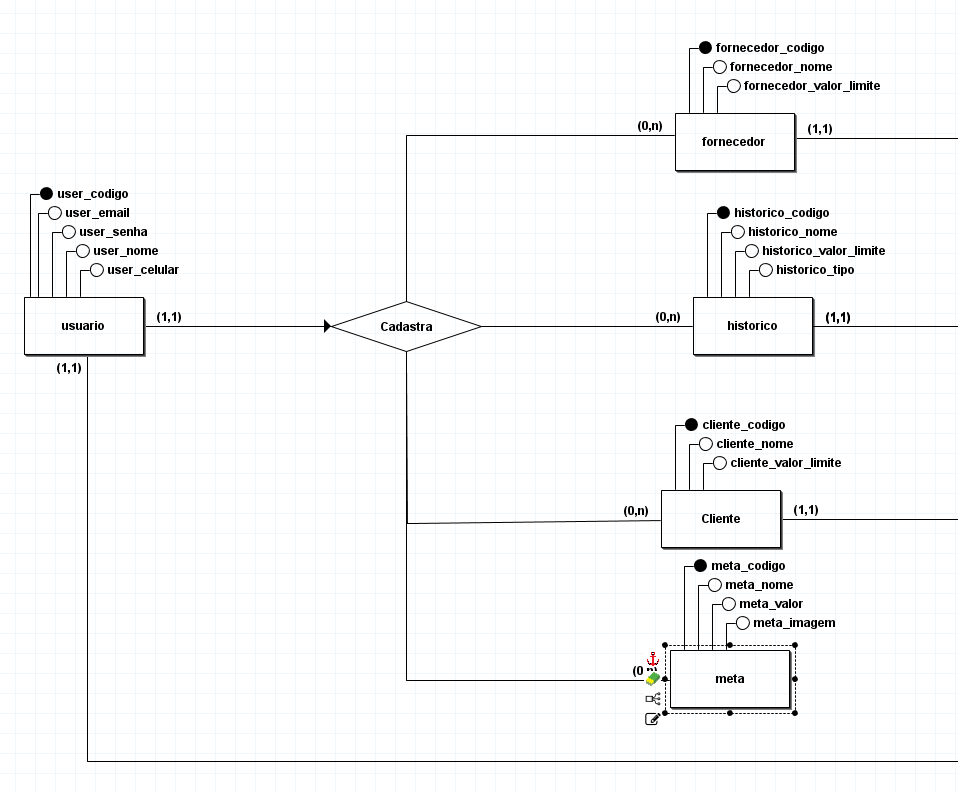


Figura - Modelo Conceitual Parte 1

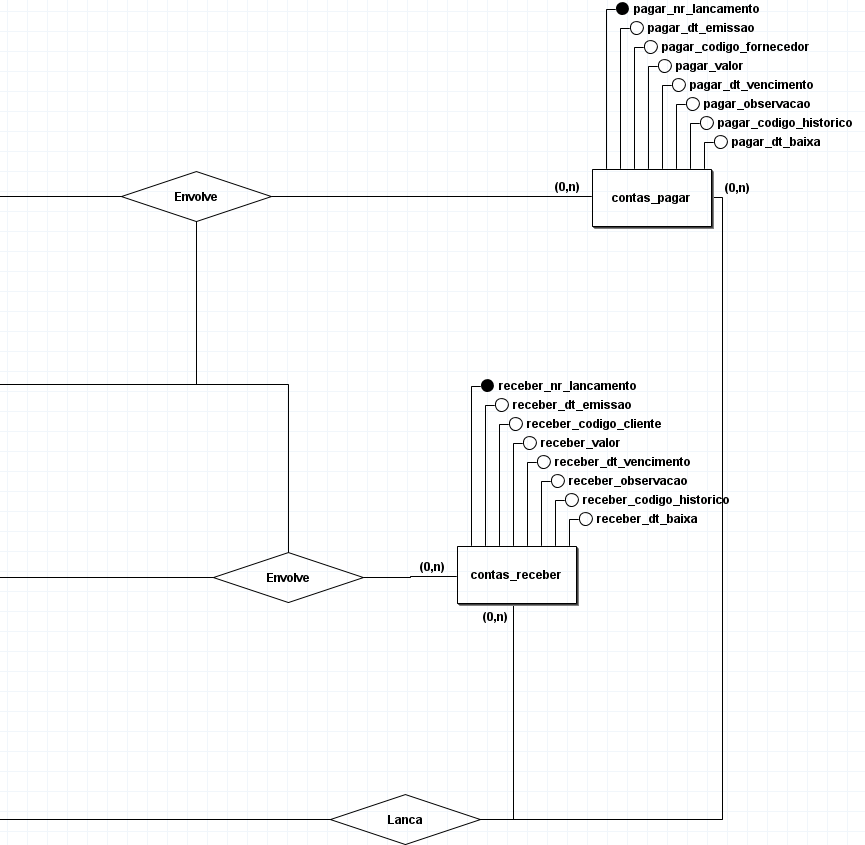


Figura - Modelo Conceitual Parte 2

# DICIONÁRIO DE DADOS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NOME** | usuario |  |  |  |
| **DESCRIÇÃO** | Tabela de usuários |  |  |  |
| **CAMPO** | **TIPO DE DADO** | **TAMANHO** | **REQUERIDO** | **COMENTÁRIO** |
| usuario\_codigo | INT |  | S | Chave Principal |
| usuario\_email | VARCHAR | 255 | S |  |
| usuario\_username | VARCHAR | 16 | S |  |
| usuario\_senha | VARCHAR | 16 | S |  |
| usuario\_nome | VARCHAR | 255 | S |  |
| usuario\_celular | INT |  | N |  |
|  |  |  |  |  |
| **NOME** | cliente |  |  |  |
| **DESCRIÇÃO** | Tabela de clientes |  |  |  |
| **CAMPO** | **TIPO DE DADO** | **TAMANHO** | **REQUERIDO** | **COMENTÁRIO** |
| cliente\_usuario | INT |  | S | Chave estrangeira relacionada à tabela usuario |
| cliente\_codigo | INT |  | S | Chave Principal |
| cliente\_nome | VARCHAR | 255 | S |  |
| cliente\_valor\_limite | DECIMAL | 10,2 | N |  |
|  |  |  |  |  |
| **NOME** | fornecedor |  |  |  |
| **DESCRIÇÃO** | Tabela de fornecedores |  |  |  |
| **CAMPO** | **TIPO DE DADO** | **TAMANHO** | **REQUERIDO** | **COMENTÁRIO** |
| fornecedor\_usuario | INT |  | S | Chave estrangeira relacionada à tabela usuario |
| fornecedor\_codigo | INT |  | S | Chave Principal |
| fornecedor\_nome | VARCHAR | 255 | S |  |
| fornecedor\_valor\_limite | DECIMAL | 10,2 | N |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
| **NOME** | tipo\_historico |  |  |  |
| **DESCRIÇÃO** | Tabela de tipos de histórico |  |  |  |
| **CAMPO** | **TIPO DE DADO** | **TAMANHO** | **REQUERIDO** | **COMENTÁRIO** |
| tipo\_codigo | INT |  | S | Chave Principal |
| tipo\_nome | VARCHAR | 255 | S |  |
|  |  |  |  |  |
| **NOME** | historico |  |  |  |
| **DESCRIÇÃO** | Tabela de históricos, históricos são as categorias dos lançamentos |  |  |  |
| **CAMPO** | **TIPO DE DADO** | **TAMANHO** | **REQUERIDO** | **COMENTÁRIO** |
| historico\_usuario | INT |  | S | Chave estrangeira relacionada à tabela usuario |
| historico\_codigo | INT |  | S | Chave Principal |
| historico\_nome | VARCHAR | 255 | S |  |
| historico\_valor\_limite | DECIMAL | 10,2 | N |  |
| historico\_tipo | INT |  | S | Chave estrangeira relacionada à tabela tipo\_historico |
|  |  |  |  |  |
| **NOME** | meta |  |  |  |
| **DESCRIÇÃO** | Tabela de metas, objetivos |  |  |  |
| **CAMPO** | **TIPO DE DADO** | **TAMANHO** | **REQUERIDO** | **COMENTÁRIO** |
| meta\_usuario | INT |  | S | Chave estrangeira relacionada à tabela usuario |
| meta\_codigo | INT |  | S | Chave Principal |
| meta\_nome | VARCHAR | 255 | S |  |
| meta\_valor | DECIMAL | 10,2 | N |  |
| meta\_imagem | VARCHAR | 255 | N |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| **NOME** | contas\_pagar |  |  |  |
| **DESCRIÇÃO** | Tabela de pagamentos |  |  |  |
| **CAMPO** | **TIPO DE DADO** | **TAMANHO** | **REQUERIDO** | **COMENTÁRIO** |
| pagar\_usuario | INT |  | S | Chave estrangeira relacionada à tabela usuario |
| pagar\_nr\_lancamento | INT |  | S | Chave Principal |
| pagar\_dt\_emissao | DATE |  | S |  |
| pagar\_codigo\_fornecedor | INT |  | S | Chave estrangeira relacionada à tabela fornecedor |
| pagar\_valor | DECIMAL | 10,2 | S |  |
| pagar\_dt\_vencimento | DATE |  | S |  |
| pagar\_observacao | VARCHAR | 255 | N |  |
| pagar\_codigo\_historico | INT |  | S | Chave estrangeira relacionada à tabela historico |
| pagar\_dt\_baixa | DATE |  | S |  |
|  |  |  |  |  |
| **NOME** | contas\_receber |  |  |  |
| **DESCRIÇÃO** | Tabela de recebimentos |  |  |  |
| **CAMPO** | **TIPO DE DADO** | **TAMANHO** | **REQUERIDO** | **COMENTÁRIO** |
| receber\_usuario | INT |  | S | Chave estrangeira relacionada à tabela usuario |
| receber\_nr\_lancamento | INT |  | S | Chave Principal |
| receber\_dt\_emissao | DATE |  | S |  |
| receber\_codigo\_cliente | INT |  | S | Chave estrangeira relacionada à tabela cliente |
| receber\_valor | DECIMAL | 10,2 | S |  |
| receber\_dt\_vencimento | DATE |  | S |  |
| receber\_observacao | VARCHAR | 255 | N |  |
| receber\_codigo\_historico | INT |  | S | Chave estrangeira relacionada à tabela historico |
| receber\_dt\_baixa | DATE |  | S |  |

# UTILIZANDO O SISTEMA

## Login

Figura - Tela de login

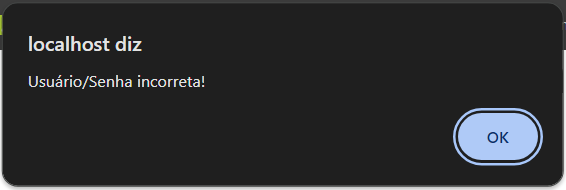
O login é feito quando já se tem um cadastro registrado no banco de dados, caso tente entrar com email/senha inexistente é apresentado um alerta do erro que ocorreu.

Figura - Erro no login

## Padrão de Cadastro

Figura - Cadastro de usuário

A página acima apresenta o padrão de todas as páginas de cadastro/edição de registros, onde usa *inputs* para receber o valor de todos os campos e os salva pelo botão “Salvar”.

## Página Inicial

Na página inicial já temos acesso ao cabeçalho de navegação que se repete em todas as demais páginas, é feita uma saudação ao usuário, é lhe apresentado o saldo que possui, e também aparecem os cartões das metas cadastradas e não concluídas. Nas metas é possível identificar o valor das mesmas junto com uma barra de progresso comparando o valor da meta com o saldo atual do usuário.

Figura - Página inicial

Ao concluir a meta, basta clicar no botão “Concluir” da respectiva meta que o cartão será removido da página inicial;

## Lista de Registros



Figura - Lista de Registros

Em cada uma das páginas presentes no cabeçalho, há uma lista com todos os registros da respectiva entidade.

O link do cabeçalho fica com cor alterada quando se está na página.

Nestas listas há o botão “Adicionar”, para acessar a página de cadastro da entidade, há um *select* onde pode-se escolher a quantidade de registros que vão aparecer em uma página da lista (caso haja mais registros do que o valor do *select* é feito paginação).

Dentro da lista, além dos dados da entidade, há também os links de ações para editar ou excluir o registro.

As únicas listas diferentes são as de Pagamentos e Recebimentos, onde, além do conteúdo que há nas demais, existe um checkbox para cada registro e um botão “Baixar”, selecionando os registros que se quer e clicando nesse botão o lançamento passa a ser contabilizado no saldo do usuário.

Figura 8 - Lista de Pagamentos com checkbox

# CONCLUSÃO

O sistema atende à sua proposta de ser uma ferramenta simplificada de organização de finanças pessoais, possui baixa complexidade e apresenta a útil funcionalidade de poder enxergar o progresso dos objetivos logo ao acessá-lo. Acredito que seja uma ferramenta funcional que pode facilmente atender às necessidades daqueles que a utilizarem para desfrutar dos benefícios de fazer um planejamento de futuros gastos.

Tirar as ideias da cabeça e levá-las a uma aplicação, bem como fazer melhorias posteriormente passaram a ser uma possibilidade plausível com a conquista de uma boa base de aprendizado advinda dos 3 anos e meio de ensino técnico e das experiências obtidas durante as tentativas e erros ao desenvolver esta e outras aplicações.

Durante a realização deste projeto de conclusão de curso, fui percebendo a importância que um bom técnico de informática tem na resolução de problemas, tanto do cotidiano, quanto na área profissional. Este é alguém que deve estar preparado com seu acervo de tecnologias (mas ainda assim disposto a aceitar e entender outras) para solucionar os problemas que lhe são apresentados com eficiência e buscando inovação de processos.

Tive que enfrentar o desafio de ir atrás de ferramentas novas, das quais eu não tinha conhecimento, para otimizar alguns pontos durante o desenvolvimento do sistema, e, em relação as tecnologias que eu conhecia, entender como aplicá-las para cada situação que aparecia até chegar à finalização do projeto. Com ele também experimentei a importância de organizar bem as ideias e o tempo, para fazer um desenvolvimento produtivo e ter um bom resultado.

# REFERÊNCIAS

How do I get the root directory path of a PHP project? Disponível em: <https://stackoverflow.com/questions/8668776/get-root-directory-path-of-a-php-project>. Acesso em: 09/05/2024.

XAMPP MySQL shutdown unexpectedly – How to Fix It. Disponível em: <https://kinsta.com/knowledgebase/xampp-mysql-shutdown-unexpectedly/>. Acesso em: 21/05/2024.

Erro - Mensagem de Login. Disponível em: <https://cursos.alura.com.br/forum/topico-erro-mensagem-de-login-63155>. Acesso em: 21/05/2024.

hash - Manual. Disponível em: <https://www.php.net/manual/pt\_BR/function.hash.php>. Acesso em: 22/05/2024.

PDOStatement::bindParam - Manual. Disponível em: <https://www.php.net/manual/pt\_BR/pdostatement.bindparam.php>. Acesso em: 01/06/2024.

Basic Branching and Merging - Git Branching. Disponível em: <https://git-scm.com/book/en/v2/Git-Branching-Basic-Branching-and-Merging>. Acesso em: 17/06/2024.

How To Get The Current URL Using JavaScript. Disponível em: <https://www.w3schools.com/howto/howto\_js\_get\_url.asp>. Acesso em: 18/06/2024.

Integração com GitHub: Implantação. Disponível em: <https://guia-de-uso.tiflux.com/integracoes/projetos/github-1/implantacao>. Acesso em: 18/06/2024.

Pagination - Bootstrap. Disponível em: <https://getbootstrap.com.br/docs/4.1/components/pagination/>. Acesso em: 20/06/2024.

Como pegar a URL anterior em JavaScript. Disponível em: <https://horadecodar.com.br/como-pegar-a-url-anterior-em-javascript/>. Acesso em: 21/06/2024.

Card - Bootstrap. Disponível em: <https://getbootstrap.com.br/docs/4.1/components/card/>. Acesso em: 24/06/2024.

CSS: Guia do Flexbox. Disponível em: <https://www.alura.com.br/artigos/css-guia-do-flexbox>. Acesso em: 26/06/2024/2024.

# FERRAMENTAS UTILIZADAS

1. Visual Studio Code – programa de edição de códigos onde há identação, tabulação, destaque de comandos e muitas outras funcionalidades, principalmente com inclusão de extensões, para facilitar o desenvolvimento;
2. Git/GitHub – ferramenta usada para gerenciar o diretório do projeto, podendo fazer versionamento do mesmo e subir o diretório para um repositório em nuvem. Link para repositório: <https://github.com/rafaelmdgh/pcc>;
3. HeideSQL -Sistema de Gerenciamento de Banco de Dados (SGBD) usado para acessar o banco de dados, bem como fazer alterações no mesmo de forma gráfica;
4. Navicat – programa usado para fazer modelos de banco de dados, conceitual, lógico e físico;
5. Br Modelo – programa usado para fazer modelo conceitual de banco de dados;
6. Xampp – facilita a instalação de um servidor local Apache para podermos fazer a conexão entre páginas web e banco de dados;

# CÓDIGO DO BANCO DE DADOS

CREATE DATABASE financas\_pessoais;

USE financas\_pessoais;

CREATE TABLE usuario  (

  usuario\_codigo int NOT NULL auto\_increment,

  usuario\_email varchar(255) NOT NULL,

  usuario\_username varchar (16) NOT NULL,

  usuario\_senha varchar(16) NOT NULL,

  usuario\_nome varchar(255) NOT NULL,

  usuario\_celular int NULL,

  PRIMARY KEY (usuario\_codigo)

) ENGINE=MYISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=1;

CREATE TABLE cliente  (

  cliente\_usuario int NOT NULL,

  cliente\_codigo int NOT NULL auto\_increment,

  cliente\_nome varchar(255) NOT NULL,

  cliente\_valor\_limite decimal(10,2) NULL,

  PRIMARY KEY (cliente\_usuario, cliente\_codigo)

) ENGINE=MYISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=1;

CREATE TABLE fornecedor  (

  fornecedor\_usuario int NOT NULL,

  fornecedor\_codigo int NOT NULL auto\_increment,

  fornecedor\_nome varchar(255) NOT NULL,

  fornecedor\_valor\_limite decimal(10,2) NULL,

  PRIMARY KEY (fornecedor\_usuario, fornecedor\_codigo)

) ENGINE=MYISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=1;

CREATE TABLE tipo\_historico  (

  tipo\_codigo int NOT NULL auto\_increment,

  tipo\_nome varchar(255) NOT NULL,

  PRIMARY KEY (tipo\_codigo)

) ENGINE=MYISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=1;

CREATE TABLE historico  (

  historico\_usuario int NOT NULL,

  historico\_codigo int NOT NULL auto\_increment,

  historico\_nome varchar(255) NOT NULL,

  historico\_valor\_limite decimal(10,2) NULL,

  historico\_tipo INT NOT NULL,

  PRIMARY KEY (historico\_usuario, historico\_codigo)

) ENGINE=MYISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=1;

CREATE TABLE meta  (

  meta\_usuario int NOT NULL,

  meta\_codigo int NOT NULL auto\_increment,

  meta\_nome varchar(255) NOT NULL,

  meta\_valor decimal(10,2) NULL,

  meta\_imagem varchar(255) NULL,

  PRIMARY KEY (meta\_usuario, meta\_codigo)

) ENGINE=MYISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=1;

CREATE TABLE contas\_pagar  (

  pagar\_usuario int NOT NULL,

  pagar\_nr\_lancamento int NOT NULL auto\_increment,

  pagar\_dt\_emissao date NOT NULL,

  pagar\_codigo\_fornecedor int NOT NULL,

  pagar\_valor decimal(10,2) NOT NULL,

  pagar\_dt\_vencimento date NOT NULL,

  pagar\_observacao varchar(255) NULL,

  pagar\_codigo\_historico int NOT NULL,

  pagar\_dt\_baixa date NOT NULL,

  PRIMARY KEY (pagar\_usuario, pagar\_nr\_lancamento)

) ENGINE=MYISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=1;

CREATE TABLE contas\_receber  (

  receber\_usuario int NOT NULL,

  receber\_nr\_lancamento int NOT NULL auto\_increment,

  receber\_dt\_emissao date NOT NULL,

  receber\_codigo\_cliente int NOT NULL,

  receber\_valor decimal(10,2) NOT NULL,

  receber\_dt\_vencimento date NOT NULL,

  receber\_observacao varchar(255) NULL,

  receber\_codigo\_historico int NOT NULL,

  receber\_dt\_baixa date NOT NULL,

  PRIMARY KEY (receber\_usuario, receber\_nr\_lancamento)

) ENGINE=MYISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=1;

ALTER TABLE cliente ADD CONSTRAINT cliente\_usuario\_codigo FOREIGN KEY (cliente\_usuario) REFERENCES usuario (usuario\_codigo) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE fornecedor ADD CONSTRAINT fornecedor\_usuario\_codigo FOREIGN KEY (fornecedor\_usuario) REFERENCES usuario (usuario\_codigo) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE tipo\_historico ADD CONSTRAINT tipo\_historico\_codigo FOREIGN KEY (tipo\_codigo) REFERENCES historico (historico\_tipo) ON DELETE RESTRICT ON UPDATE CASCADE;

ALTER TABLE historico ADD CONSTRAINT historico\_usuario\_codigo FOREIGN KEY (historico\_usuario) REFERENCES usuario (usuario\_codigo) ON UPDATE CASCADE ON DELETE CASCADE;

ALTER TABLE meta ADD CONSTRAINT meta\_usuario\_codigo FOREIGN KEY (meta\_usuario) REFERENCES usuario (usuario\_codigo) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE contas\_pagar ADD CONSTRAINT contas\_pagar\_usuario\_codigo FOREIGN KEY (pagar\_usuario) REFERENCES usuario (usuario\_codigo) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE contas\_pagar ADD CONSTRAINT contas\_pagar\_fornecedor\_codigo FOREIGN KEY (pagar\_codigo\_fornecedor) REFERENCES fornecedor (fornecedor\_codigo) ON DELETE RESTRICT ON UPDATE CASCADE;

ALTER TABLE contas\_pagar ADD CONSTRAINT contas\_pagar\_historico\_codigo FOREIGN KEY (pagar\_codigo\_historico) REFERENCES historico (historico\_codigo) ON DELETE RESTRICT ON UPDATE CASCADE;

ALTER TABLE contas\_receber ADD CONSTRAINT contas\_receber\_usuario\_codigo FOREIGN KEY (receber\_usuario) REFERENCES usuario (usuario\_codigo) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE contas\_receber ADD CONSTRAINT contas\_receber\_cliente\_codigo FOREIGN KEY (receber\_codigo\_cliente) REFERENCES cliente (cliente\_codigo) ON DELETE RESTRICT ON UPDATE CASCADE;

ALTER TABLE contas\_receber ADD CONSTRAINT contas\_receber\_historico\_codigo FOREIGN KEY (receber\_codigo\_historico) REFERENCES historico (historico\_codigo) ON DELETE RESTRICT ON UPDATE CASCADE;

insert into tipo\_historico (tipo\_nome) values ("Receitas");

insert into tipo\_historico (tipo\_nome) values ("Despesas Fixas");

insert into tipo\_historico (tipo\_nome) values ("Despesas Diversas");

insert into tipo\_historico (tipo\_nome) values ("Investimentos");